

# Project Report

**Title of Project:**

Orbit Lens: 3D Real-Time Tracker

**Name of the Innovator:**

Vedant Sutar

**Start Date:**

11/02/2026

**End Date:**

18/02/2026

## Day 1: Empathise & Define

### Step 1: Understanding the Need

#### Which problem am I trying to solve?

Existing satellite tracking tools are often desktop-only, visually outdated, or too complex for casual users to understand. There is no easy way to visualize satellite orbits in 3D on a mobile device.

#### Who is affected by this problem?

Space enthusiasts, students, amateur astronomers, and educators who want to teach orbital mechanics.

#### How did I find out about this?

Observation, Online Research, AI Tools

### Step 2: Problem Statement

Space enthusiasts and students lack a modern, mobile-responsive, and visually intuitive tool to track real-time satellite positions, making space science feel inaccessible.

## **Why is this problem important to solve?**

Making space data accessible helps inspire the next generation of scientists. Real-time visualization helps people connect with the technology orbiting above them.

### **Take-home task insights:**

Users prioritize mobile usability and "cool factor" (visuals) over raw data tables. The interface needs to feel like a sci-fi "Mission Control".

## **Day 2: Ideate**

### **Step 3: List at least 5 different solutions:**

1. A text-based SMS alert system for ISS flyovers.
2. A 2D flat-map website with satellite icons.
3. A 3D interactive Globe Web App with real-time physics.
4. An Augmented Reality (AR) mobile app to point at the sky.
5. A daily email newsletter with satellite locations.

### **Step 4: My favourite solution:**

Solution 3 (3D Interactive Globe Web App).

### **Step 5: Why am I choosing this solution?**

It offers the best balance of visual engagement and accessibility. A web app works on all devices without installing software, and a 3D globe is the most accurate way to understand orbits.

## **Day 3: Prototype & Test**

### **Step 6: What will my solution look like?**

A "Dark Mode" web application featuring a central 3D Earth. It uses the satellite.js library to calculate positions in real-time. The mobile view features a "swipe-up" telemetry card to save screen space.

## **What AI tools will I need?**

Code generation for React components, logic assistance for orbital math (TLE conversion), and UI design for responsive layouts.

### **Selected AI tools:**

1. Lovable (Full-stack AI Builder)
2. React & Tailwind CSS (Frontend)
3. Satellite.js (Orbital Math Library)
4. Supabase (Backend/Data)
5. Lucide React (Icons)

## **Step 7: Test - Getting Feedback**

### **Who did I share my solution with?**

Self-testing on mobile and desktop; peer review for UI feedback.

### **What works well:**

The "Mobile-First" design is successful—the globe looks great on small screens, and the bottom-sheet data display keeps the view uncluttered.

### **What needs improvement:**

The 3D globe texture takes a few seconds to load on slower mobile networks, causing a brief blank screen. The 'Search' bar keyboard sometimes covers the satellite data when typing on smaller phones. We also need to add more satellite categories (like Weather or GPS) beyond just Starlink and ISS.

## **Day 4: Showcase**

## **Step 8: Final Project Title:**

Orbit Lens: 3D Real-Time Tracker

## **1-Minute Pitch Summary:**

OrbitLens brings a NASA-grade Mission Control experience to your pocket. While most trackers are clunky and hard to use, OrbitLens uses a high-fidelity 3D globe to visualize the ISS and Starlink constellations in real-time. We solved the challenge of displaying complex orbital math on mobile devices by using a 'bottom-sheet' UI design. It turns abstract data into a beautiful, educational experience that anyone can access instantly.

## **Step 9: Reflections**

### **What did I enjoy the most?**

Seeing the math actually work—watching the satellite path draw itself across the globe in real-time was very satisfying.

### **What was my biggest challenge?**

Optimizing the 3D globe performance for mobile phones and ensuring the UI elements (buttons/text) didn't cover the Earth on small screens.

### **Project Link:**

<https://orbit-watcher-station.lovable.app>